

68. Despite my decision to employ only a fixed menu structure for the product as a whole, during this time I continued to refine my ideas for user-definable menus. For many months I considered including a "command language" in the product, which would enable the user to automate repetitive command sequences. This idea ultimately resulted in the development and incorporation in 1-2-3 of keystroke "macros". A macro permits a user to store a sequence of keystrokes which the user can later instruct the computer to execute by invoking a single command.

69. I was aware of the concept of macros from prior experiences with text editor programs. To my knowledge, no other spreadsheet at the time expressed the macro idea the way we did in 1-2-3.

70. Although there was much give-and-take on the desirability of including macro capabilities, after lengthy experimentation I ultimately decided to do so in the belief that sophisticated users would find them valuable.

71. We also considered the selection of commands to provide to the user for construction and operation of macros. The universe of possible commands was very broad, some of which could present considerable complexity. Ultimately, we determined to include a set of eight commands that we believed would be relatively easy for users of some sophistication to grasp, but [\*] would also enhance the functionality of the macro feature to a meaningful degree.

#### *Development of 1-2-3; Menu Structure*

72. The essential functionality the product would provide was settled approximately in August 1982. Our primary focus then turned to the precise iteration of the menu tree.

73. From August until about October 1982, I met with Sachs and the other members of our development team, including John Posner, Ezra Gottheil, Mary Goldschmidt and Steve Miller, several times a week to determine each word choice, location and menu arrangement.

74. On my own and collectively with this team, hundreds of hours were spent refining the menus, and dozens of iterations of menus and their organization were proposed. The process induced intense debate at times. In the end, the final decisions were mine.

75. In selecting words for the menus, we struggled to choose words that would intelligently convey to the user the purpose of each command and its underlying functionality, while at the same time ensuring that each word in a particular menu level began with a unique letter. The first concern was based on our desire to make the menus as informative and intuitive as possible. The latter concern derived from the previous design choice to permit the user to invoke menu commands by striking the key corresponding to the first letter of each word; obviously, if [\*] more than one word began with the same first letter, the program could not determine which command the user intended by striking the key corresponding to both.

76. Another factor we considered in organizing the menus was my belief—based upon the principle of cognitive psychology that individuals can retain only seven pieces of information (plus or minus two) at a time—that each menu level should contain approximately seven items (or fewer), to be easier to learn and use.

77. I also believed that the hierarchical arrangement of the menus should reflect a structured approach that communicated the product's underlying functionality to the user. For example, I felt that commands representing related features should be grouped together on a single level of the menu tree, with subsequent dependent commands appearing on lower levels of the tree.

78. In addition, in organizing the menus I believed that the commands most likely to be used frequently should be grouped in a broad menu appearing at the top of the menu structure. Similarly, I felt that the commands displayed within a given menu level should appear from left to right in declining order of likely frequency of usage.

79. Because there was no precedent for the menu structure we envisioned, we had no formal market research or other data to guide us in making these determinations. We did, [\*] however, conduct a number of informal user tests both internally at Lotus as well as outside the company. In the end, the menu tree organization was based largely on my intuition and subjective judgment, informed by feedback from the user tests, trying as best I could to imagine myself in the role of a typical user.

80. The numerous possibilities considered in creating the final Lotus 1-2-3 menu structure are exemplified in my contemporaneous notes of May 29, July 24, and August 18, 1982 (Exhibits I, J, and K respectively), which depict the proposed menu structure at various stages of development.

81. Attached as Exhibits L through O to this affidavit are true and correct copies of documents maintained in my files created on or about August 4, August 17, September 22, and October 4, 1982, respectively, which I recognize as reflecting the Lotus 1-2-3 menu structure as it existed on or about those dates.

82. As Exhibits I through O reflect, there were several possible choices for almost every word selected, as well as for its location in the menu trees. For example, the "Copy" command could also have been called "Replicate", "Duplicate", "Repeat", or "Clone".

83. One example of an instance where both the word used to represent a command, and its location in the menu tree, underwent considerable change in the course of development is that of the "Range" command and the commands branching from it.

84. Its origin was an idea I had in April 1982 to [\*] include a "named range" feature, whereby the user could refer to ranges of cells in the worksheet by assigning them user-defined names (in addition to referencing them by other means). I thought this feature, not present in any existing product, would be useful for specifying commonly referred-to

ranges on the worksheet, including tables, data to be graphed, and areas to be printed.

85. I originally contemplated implementing this feature through a command called "Name" located on the top menu level, which, when invoked, would display all named ranges the user had previously specified horizontally in the control panel. A user could select a desired named range by moving the cursor to the applicable name and striking the "Enter" key, which would then invoke a submenu containing the following options: Accept, Exit, Edit, Rename, Delete, and Cancel.

86. Throughout the summer I struggled to simplify and refine the "named range" feature. One alternative I considered was to assign a "Name" command to a function key, either instead of or in addition to a "Name" menu command.

87. Finally, following numerous meetings and discussions, in August I decided to replace "Name" in the top menu level with "Range", and to locate a "Name" menu command—permitting the user to create, modify or delete named ranges—in the branching submenu. The feature permitting the user to select previously-specified named ranges from a horizontal display was [\*] assigned to a "Name" function key. In September, the command options for the submenu branching further from the "Range Name" commands were selected.

88. During the intensive phase of work on the menus from August to October 1982, another debate erupted regarding the "Worksheet Global" command options. These commands were intended to permit the user to set certain parameters for what we called the "worksheet", including the format, recalculation mode, recalculation order and current and default column widths.

89. These commands initially branched from a command called "Set" located in the top menu level. I became dissatisfied with this structure because I did not believe it gave sufficient emphasis to the "global" or worksheet-wide nature of most of the commands; at the same time, some of the com-

mands following "Set" were not, originally, of that same "global" nature.

90. My initial thought was to replace the "Set" command with a "Global" command option in the top menu. That, however, created an initial-letter conflict because the word "Graph" already appeared in that menu.

91. I had long discussions with my development team regarding alternative words to represent the "Global" command. One possibility suggested was "Default". Again, however, we already had chosen the word "Data" to represent the database branches of the menu tree in the top menu.

[\*]

92. I ultimately decided that "Global" was the most descriptive and appropriate word. To resolve the initial-letter conflict with "Graph", I moved "Global" from the top menu to a submenu branching from the "Worksheet" command.

93. A related decision was to reorganize the features permitting the user to set current and default (or worksheet-wide) column widths, both of which originally branched from the "Global" command. I felt this was potentially confusing to the user. The feature allowing the user to set the width solely for the column he or she was then working in (the "current" column) did not logically fit with the rest of the "Global" commands.

94. To provide greater clarity and to present a better-structured approach in the menu tree, I broke these features apart, placing one "Column-Width" command in the submenu branching from "Global", to represent the capability to alter widths for all columns in the worksheet, and another "Column-Width" command in the same submenu as "Global", to represent the feature pertaining to the current column only. That way, the user would be unlikely to mistake the effect of a column-width adjustment, since the user would have to select the "Global" command only in instances when the user desired to make a global change.

95. It also became necessary to change the sequence of the "Global" command submenus as Sachs implemented new [\*] functionality. For example, when Sachs initially added the "Iteration" and "Recalculation" features, they both appeared in the submenu directly branching from the "Worksheet Global" commands. Because of a shortage of space on the screen in that menu level, the word "Recalculation" was abbreviated to "Recalc".

96. Ultimately, however, as Sachs continued to refine the program's functionality, we decided to move "Iteration" down one menu level in the sequence, to branch from the "Recalc" command. Once this change was made and additional space was available, I restored "Recalc" to its original form—"Recalculation".

97. We also confronted difficulty in choosing a word to represent the feature that permitted a user to save a selected range of cells (including formulas), rather than an entire worksheet, in a separate file. Originally, and throughout the summer of 1982, this feature was represented by the term "File Extract".

98. In September, when we changed the "File Delete" command option to "File Erase" (to avoid the conflict that arose when we renamed the "File Current" command "File Disk" and, ultimately, "File Directory"), we encountered yet another initial-letter conflict, since it was not possible to have both "Erase" and "Extract" on the same menu level. We could not find an acceptable alternative for "Erase"; thus, we had to reconsider "Extract".

[\*]

99. One suggestion was "Partial". As is reflected in Exhibit N, I did not think "Partial" was a satisfactory alternative, because it was not sufficiently informative. One of the members of the team suggested using the non-English word "Xtract", so we could convey the desired information without conflicting keystrokes. "Xtract" is, in fact, the alternative that we ultimately used.



100. In addition to selecting the words to represent each command and determining their organization in the menu tree, from August to October 1982 we also created the corresponding long prompts displayed in the lower line of the menus. Each of the above-mentioned team members participated.

101. In some instances, we wrote brief textual explanations for the commands appearing in the menus. In others, we chose to display the next submenu branching from a particular command. Our decisions in this regard were governed not by a mechanical principle but by a case-by-case consideration as to which type of long prompt would be appropriate to a particular command, with the ultimate goal of providing better information to the user.

\* \* \*

[\*]

112. In September 1985, Lotus released a new version or upgrade of Lotus 1-2-3, called Release 2.0. I was involved in the development of Release 2.0 primarily in the roles of product design and project management and oversight.

113. Release 2.0's enhancements to the features provided in Release 1A resulted in the addition of a number of commands, and necessitated some changes to the existing menus. For example, the addition of the "Value" and "Transpose" commands to the submenu branching from the "Range" command required the abbreviation of the words "Label-Prefix" appearing in Release 1A to the single word "Label" in Release 2.0, to represent the same command. The addition of a "Hidden" command to the submenu branching from the "Range Format" commands led to a slight change in the order of items within that submenu.

114. However, the user interface of 1-2-3, Release 1A, expanded and modified as necessary to express new capabilities, was otherwise essentially incorporated in Release 2.0.

Signed under the penalties of perjury this 27th day of November, 1989.

/s/ MITCHELL D. KAPOR  
Mitchell D. Kapor

# **OPPOSITION BRIEF**

4  
No. 94-2003

Supreme Court, U. S.

FILED

JUL 6 1995

CLERK

# In the Supreme Court

OF THE

## United States

OCTOBER TERM, 1994

LOTUS DEVELOPMENT CORPORATION,  
*Petitioner,*

v.

BORLAND INTERNATIONAL, INC.  
*Respondent.*

On Petition for a Writ of Certiorari to  
the United States Court of Appeals  
for the First Circuit

### BRIEF IN OPPOSITION TO PETITION FOR CERTIORARI

GARY L. REBACK  
(Counsel of Record)  
MICHAEL BARCLAY  
ISABELLA E. FU  
WILSON, SONSINI,  
GOODRICH & ROSATI  
Professional Corporation  
650 Page Mill Road  
Palo Alto, California 94304  
Telephone: (415) 493-9300

PETER E. GELHAAR  
KATHERINE L. PARKS  
DONNELLY, CONROY &  
GELHAAR  
One Post Office Square  
33rd Floor  
Boston, Massachusetts 02109  
Telephone: (617) 542-7400

*Attorneys for Respondent*

3844

**QUESTION PRESENTED**

Did the First Circuit correctly hold that commands used to operate a computer spreadsheet program are uncopyrightable under 17 U.S.C. § 102(b)?

## RULE 29.1 STATEMENT

Respondent Borland International, Inc. has no parent corporation or subsidiaries that are not wholly owned, except for certain foreign subsidiaries in which a minimal amount of shares (fewer than 1%), which are not publicly traded, are held by foreign nationals in accordance with local law.

## TABLE OF CONTENTS

	<u>Page</u>
QUESTION PRESENTED .....	i
RULE 29.1 STATEMENT .....	ii
TABLE OF AUTHORITIES .....	v
INTRODUCTION AND SUMMARY .....	1
STATEMENT OF THE CASE .....	7
A. Factual Background .....	8
1. The Lotus Product .....	8
2. The Borland Product .....	11
B. Proceedings in the District Court .....	11
REASONS FOR DENYING THE PETITION .....	13
I. THE CIRCUITS ARE IN AGREEMENT THAT WORDS USED AS BUTTONS TO OPERATE A PROGRAM, UNLIKE THE PROGRAM ITSELF, ARE UNCOPYRIGHTABLE .....	13
A. The District Court Erroneously Applied the "Abstractions" Test to Uncopyrightable Subject Matter .....	13
B. The District Court's Decision to Extend Copyright to the Words That Operate the Program Ran Afoul of Both the Statute and the Case Law .....	17



## TABLE OF CONTENTS (Cont'd)

	<u>Page</u>
C. There Is No Conflict Among the Circuits That Menu Commands and Similar Methods of Operation Are Uncopyrightable .....	20
II. THE FIRST CIRCUIT OPINION REMOVES THE UNCERTAINTY PRODUCED BY THE DISTRICT COURT AND UPHOLDS THE OVERALL INTELLECTUAL PROPERTY FRAMEWORK ESTABLISHED BY CONGRESS .....	24
A. The First Circuit Opinion Restores Clarity and Predictability to the Law .....	24
B. Both Copyright and Patent Remain Sources of Strong Protection, Providing Enormous Incentives for Authors and Inventors of Software Products to Innovate .....	26
CONCLUSION .....	27

## TABLE OF AUTHORITIES

	<u>Page(s)</u>
<b>CASES</b>	
<i>Ashton-Tate Corp. v. Ross</i> , 728 F. Supp. 597 (N.D. Cal. 1989) .....	21
<i>Ashton-Tate Corp. v. Ross</i> , 916 F.2d 516 (9th Cir. 1990) .....	6, 20, 21, 24
<i>Autoskill, Inc. v. National Educ. Support Sys., Inc.</i> , 994 F.2d 1476 (10th Cir.), cert. denied, 114 U.S. 307 (1993) .....	21, 22
<i>Baker v. Selden</i> , 101 U.S. 99 (1879) .....	passim
<i>Brown Bag Software v. Symantec Corp.</i> , 960 F.2d 1465 (9th Cir.), cert. denied sub. nom. <i>BB Asset Mgmt. Inc. v. Symantec Corp.</i> , 113 S. Ct. 198 (1992) .....	20
<i>Computer Assoc. Int'l, Inc. v. Altai, Inc.</i> , 982 F.2d 693 (2d Cir. 1992) .....	passim
<i>Digital Comm. Assoc., Inc. v. Softklone Distrib. Corp.</i> , 659 F. Supp. 449 (N.D. Ga. 1987) .....	22
<i>Engineering Dynamics, Inc. v. Structural Software, Inc.</i> , 26 F.3d 1335 (5th Cir. 1994) .....	7, 23
<i>Engineering Dynamics, Inc. v. Structural Software, Inc.</i> , 46 F.3d 408 (5th Cir. 1995) .....	7, 23
<i>Gates Rubber Co. v. Bando Chem. Indus., Ltd.</i> , 9 F.3d 823 (10th Cir. 1993) .....	6, 15, 16, 22

## TABLE OF AUTHORITIES (Cont'd)

	<u>Page(s)</u>
<i>Gates Rubber Co. v. Bando Chem. Indus., Ltd.</i> , No. 92-S-136 (D. Colo. filed June 12, 1995) ...	6, 22
<i>Johnson Controls, Inc. v. Phoenix Control Sys., Inc.</i> , 886 F.2d 1173 (9th Cir. 1989) .....	20
<i>Lotus Dev. Corp. v. Borland Int'l, Inc.</i> , 788 F. Supp. 78 (D. Mass. 1992) .....	passim
<i>Lotus Dev. Corp. v. Borland Int'l, Inc.</i> , 799 F. Supp. 203 (D. Mass. 1992) .....	passim
<i>Lotus Dev. Corp. v. Borland Int'l, Inc.</i> , 831 F. Supp. 202 (D. Mass. 1993) .....	2, 10
<i>Lotus Dev. Corp. v. Borland Int'l, Inc.</i> , 831 F. Supp. 223 (D. Mass. 1993) .....	2, 10
<i>Lotus Dev. Corp. v. Borland Int'l, Inc.</i> , 49 F.3d 807 (1st Cir. 1995) .....	passim
<i>Lotus Dev. Corp. v. Paperback Software Int'l</i> , 740 F. Supp. 37 (D. Mass. 1990) .....	passim
<i>Manufacturers Technologies, Inc. v. CAMS, Inc.</i> , 706 F. Supp. 984 (D. Conn. 1989) .....	22
<i>Mitek Holdings, Inc. v. Arce Eng'g Co.</i> , 864 F. Supp. 1568 (S.D. Fla. 1994) .....	23
<i>Sony Corp. v. Universal City Studios, Inc.</i> , 464 U.S. 417 (1984) .....	27

## TABLE OF AUTHORITIES (Cont'd)

	<u>Page(s)</u>
<i>Whelan Assoc., Inc. v. Jaslow Dental Laboratory, Inc.</i> , 797 F.2d 1222 (3d Cir. 1986), cert. denied, 479 U.S. 1031 (1987) .....	14, 15, 23, 24

## STATUTES, RULES, AND REGULATIONS

60 Fed. Reg. 28,778 (proposed June 2, 1995) .....	27
H.R. Rep. No. 1476, 94th Cong., 2d Sess. 54-55 (1976), reprinted in 1976 U.S.C.C.A.N. 5659 ...	14
17 U.S.C. § 101 .....	13
17 U.S.C. § 102(b) .....	passim
28 U.S.C. § 1292(a) .....	12

## MISCELLANEOUS

Copyright Office Circular 61, <i>Copyright Registration for Computer Programs</i> .....	25
Paul Goldstein, <i>Copyright</i> (1989) .....	14, 25
Paul Goldstein, <i>Infringement of Copyright in Computer Programs</i> , 47 U. Pitt. L. Rev. 1119 (1986) .....	25
Karen S. Kovach, Comment, <i>Computer Software Design: User Interface — Idea or Expression?</i> 60 U. Cin. L. Rev. 161 (1991) .....	3

## TABLE OF AUTHORITIES (Cont'd)

	<u>Page(s)</u>
Kenneth A. Liebman, <i>et al.</i> , <i>Back To Basics: A Critique Of The Emerging Judicial Analysis Of The Outer Limits Of Computer Program Expression</i> , 2 Computer Law., Dec. 1985, at 1 . . .	15
Steven W. Lundberg <i>et al.</i> , <i>Identifying Uncopyrightable Computer Implemented Processes and Systems</i> , 9 Computer Law., Apr. 1992, at 7 . . . . .	3
David Nimmer & Melville B. Nimmer, <i>Nimmer on Copyright</i> (1994) . . . . .	13
Pamela Samuelson, <i>Computer Programs, User Interfaces, and Section 102(b) of the Copyright Act of 1976: A Critique of Lotus v. Paperback</i> , 55 Law & Contemp. Probs., Spring 1992, at 311	3
Timothy S. Teter, Note, <i>Merger and the Machines: An Analysis of the Pro-Compatibility Trend in Computer Software Copyright Cases</i> , 45 Stan. L. Rev. 1061 (1993) . . . . .	3
U.S. Department of Commerce, Press Release No. 95-18, <i>USPTO to Develop Guidelines to Protect Software Inventions</i> , March 30, 1995 . . . . .	27
Julian Velasco, <i>The Copyrightability of Nonliteral Elements of Computer Programs</i> , 94 Columbia L. Rev. 242 (1994) . . . . .	3

## TABLE OF CONVENTIONS

Borland	—	Respondent Borland International, Inc.
Lotus	—	Petitioner Lotus Development Corporation
Petition or Pet.	—	Lotus' Petition for a Writ of Certiorari filed in this Court
Pet. App.	—	Appendix to Lotus' Petition filed in this Court
1st Cir. App.	—	Joint Appendix of Record filed in the U.S. Court of Appeals for the First Circuit (the court below)
Borland S.J. Brief	—	Borland's Memorandum in Support of Cross-Motion for Summary Judgment (Dkt. No. 141) in the district court
Ex.	—	Exhibit ____ from Exhibits in Support of Borland's Cross-Motion for Summary Judgment (Dkt. No. 142) in the district court
RE	—	Excerpts of Record attached to Borland's opening brief in the court below (filed Dec. 14, 1993)

No. 94-2003

# In the Supreme Court

OF THE

**United States**

---

OCTOBER TERM, 1994

---

LOTUS DEVELOPMENT CORPORATION,  
*Petitioner,*

v.

BORLAND INTERNATIONAL, INC.  
*Respondent.*

---

**On Petition for a Writ of Certiorari to  
the United States Court of Appeals  
for the First Circuit**

---

**BRIEF IN OPPOSITION  
TO PETITION FOR CERTIORARI**

---

## INTRODUCTION AND SUMMARY

The Court of Appeals in this case held that the commands used to operate a computer spreadsheet program — common commands such as “COPY,” “MOVE,” or “PRINT” displayed to the user on a computer screen—are uncopyrightable under § 102(b) of the Copyright Act. The court’s opinion relied on the express language of the statute that forbids copyright protection for “methods of operation” and “systems.” The court also relied on this Court’s seminal decision in the area, *Baker v. Selden*, 101 U.S. 99 (1879), which mandates that claims for the protection of methods of operation and systems be governed by the *patent* law rather than the *copyright* law.



The First Circuit's opinion carefully reviewed the leading authority from the other circuits, principally the Second Circuit's decision in *Computer Assoc. Int'l, Inc. v. Altai, Inc.*, 982 F.2d 693 (2d Cir. 1992), that sets forth a methodology for evaluating claims of copyright infringement in the text (i.e., "code") and "structure" of computer programs. *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 49 F.3d 807, 814 (1st Cir. 1995), Pet. App. at 13a-15a. The First Circuit readily agreed that the methodology used in *Altai* and similar cases provides "a useful framework" for evaluating infringement claims in "code" and "code structure" but was simply inapposite to the issue here—the copyrightability of command words, or "menus," used to operate the program. 49 F.3d at 815, Pet. App. at 14a. The First Circuit viewed this issue as easily resolvable by reference to the statute and Supreme Court authority.

Judge Boudin filed a separate opinion in which he concurred in the majority's reasoning as well as its conclusion. 49 F.3d at 821, Pet. App. at 27a-28a. Judge Boudin went on to explain that extending copyright protection to the command words at issue by *judicial fiat*, as the district court had done, is at variance with the intent of Congress and is both inefficient and anticompetitive from an economic perspective. *Lotus* should look to the patent law, rather than copyright, to protect its method of operation.

The First Circuit decision reversed a series of opinions in this case from a single judge in the District of Massachusetts.<sup>1</sup> From the beginning, the district court eschewed a narrow focus on what was actually at issue in this case—the method by which the user tells the computer program what to do. Instead, the district court

<sup>1</sup>See *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 788 F. Supp. 78 (D. Mass. 1992) ("*Borland I*"), Pet. App. at 145a; *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 799 F. Supp. 203 (D. Mass. 1992) ("*Borland II*"), Pet. App. at 106a; *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 831 F. Supp. 202 (D. Mass. 1993) ("*Borland III*"), Pet. App. at 71a; *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 831 F. Supp. 223 (D. Mass. 1993) ("*Borland IV*"), Pet. App. at 29a. Each of these opinions refers to and is based upon an earlier decision of the district court, *Lotus Dev. Corp. v. Paperback Software Int'l*, 740 F. Supp. 37 (D. Mass. 1990) ("*Paperback*"), Pet. App. at 183a, which decision was not appealed.

viewed the case as the opportunity to involve itself in the developing case law regarding the protection of computer program "code" and "structure." Although those issues are *not* present in this case, the district court sought to inject this case into the debate about those issues, by creating its own novel methodology to determine the copyrightability of all aspects of computer technology.

The *Lotus* petition describes the district court opinions variously as "virtually canonical," as having "a prominent role" and as a "touchstone" in computer copyright law. Petition at 2, 15, 22. Exactly the opposite is true. The district court decisions in this case and in the predecessor *Paperback* case provoked a firestorm of controversy. As the record below reflects, the district court opinions were widely criticized in the academic community,<sup>2</sup> the legal press,<sup>3</sup> the national financial press,<sup>4</sup> and the computer

<sup>2</sup>See, e.g., Steven W. Lundberg et al., *Identifying Uncopyrightable Computer Implemented Processes and Systems*, 9 Computer Law., Apr. 1992, at 7, 9 ("the Court in *Lotus* [*v. Paperback*] could never have reached the correct conclusion since it never laid the fundamental groundwork for it"); Timothy S. Teter, Note, *Merger and the Machines: An Analysis of the Pro-Compatibility Trend in Computer Software Copyright Cases*, 45 Stan. L. Rev. 1061 (1993); Pamela Samuelson, *Computer Programs, User Interfaces, and Section 102(b) of the Copyright Act of 1976: A Critique of Lotus v. Paperback*, 55 Law & Contemp. Probs., Spring 1992, at 311, 352-53; Karen S. Kovach, Comment, *Computer Software Design: User Interface—Idea or Expression?* 60 U. Cin. L. Rev. 161 (1991) (*Paperback* improperly extended copyright protection to *Lotus*' menu command system); Julian Velasco, *The Copyrightability of Nonliteral Elements of Computer Programs*, 94 Columbia L. Rev. 242, 262-65 (1994). These authorities are cited in *Borland's* Petition for Initial In Banc Hearing, filed with the First Circuit on November 10, 1993 ("*Borland's In Banc Brief*"), at 2 n.9.

<sup>3</sup>See, e.g., Susan Kostal, *Copyright Scholars Want a Fine Point Put on Spreadsheet Case*, S.F. Daily J., Oct. 4, 1991, at 1, 8; *Lotus Wins Copyright Suit*, Mass. Law. Wkly., Aug. 10, 1992, at 23. See *Borland's In Banc Brief* at 3 n.10.

<sup>4</sup>See, e.g., *Borland Gains in Bid For Appeals Ruling On Lotus Copyrights*, Wall. St. J., Aug. 20, 1993, at B3; William M. Bulkeley,

industry press<sup>5</sup> for extending copyright protection to put large sectors of the software industry off limits to competition, thereby producing a radical departure from the leading authority of other circuits. Although Lotus claims that the Second Circuit's *Altai* decision cites the district court opinions at issue here "with approval," Pet. at 16, in fact the Second Circuit *specifically cited* and pointedly *rejected* the district court's "incentive-based arguments in favor of broad copyright protection" as having a "corrosive effect on certain fundamental tenets of copyright doctrine." 982 F.2d at 712. Astonishingly, the Lotus petition fails to disclose the Second Circuit's pointed criticism of the district court's position.

In the proceedings below, a myriad of industry associations (representing both large and small companies), consumer groups, distinguished academics, eminent computer scientists, and even the Register of Copyrights of the United States all filed amicus briefs criticizing the district court and urging reversal of its opinions.<sup>6</sup> Lotus' position, by contrast, has attracted little support

---

*Borland Case Briefs Raise Questions About Software Copyright Protection*, Wall. St. J., Oct. 4, 1991, at B4; William M. Bulkeley, *Borland Loses Infringement Claim by Lotus*, Wall St. J., Aug. 13, 1993, at B5; T.R. Reid, *Consumers Lose When Software Makers Sue Each Other*, Wash. Post, Oct. 25, 1993, at F18; John R. Wilke, *Ruling Against Borland May Intensify Copyright Debate*, Wall. St. J., August 3, 1992, at B1, B4. See Borland's In Banc Brief at 3 n.11.

<sup>5</sup>Heather Clancy, *Copyright laws blurred*, Computer Reseller News, Aug. 17, 1992, at 137; Peter Coffee, *Key Reader verdict shows need for open standards*, PC Week, Aug. 30, 1993, at 32; Ed Foster, *That's a nice macro you wrote. Too bad Lotus now owns it*, InfoWorld, Aug. 23, 1993, at 45; Ed Foster, *Too-tight legal controls can straightjacket a whole industry*, InfoWorld, Sept. 6, 1993, at 49; Beth Freedman, *Experts Question Ruling In Lotus-Borland Case*, PC Week, Aug. 10, 1992, at 6; Steve Gibson, *Lotus ruling will damage the industry if it protects languages*, InfoWorld, Sept. 14, 1992, at 42. See Borland's In Banc Brief at 3 n.12.

<sup>6</sup>See the following: Brief Amicus Curiae of Computer Software Industry Association, representing over 3000 companies and professional individuals; Brief of Amicus Curiae Software Entrepreneurs' Forum,

in the *software* industry. Its amici have consisted of a few large companies who favor the extension of copyright law to inhibit competition from the hundreds of other operating system and application software companies.

Lotus' argument that the First Circuit decision created a conflict in the circuits is pure fabrication. Lotus argues that there is a consensus in the circuits regarding the application of the "abstractions" test to computer programs, that the district court's opinions in this case are a part of that consensus, and that the First Circuit's decision departs from that body of law. But, in fact, the discussion in the circuit courts regarding the "abstractions" test is directed to different subject matter—*i.e.*, the program code and structure—rather than the menus used to operate the program. That is the principal failing in Lotus' argument and the reason Lotus lost this case. As the First Circuit made plain, this case is *not* about a computer program; it is about the menu words that are used as buttons and switches to operate the program. The First Circuit decision does not stand for the proposition that it is error to apply the "abstractions" test *to computer programs*. Rather, the First Circuit has held that it is error to apply the "abstractions" test to something (in this case menus) that is not copyrightable in the first place.

Nor did the First Circuit reject the "idea/expression" dichotomy in applying § 102(b) as Lotus asserts. Pet. at 23. Rather, referring to the express language of the statute, the First Circuit recognized that § 102(b) makes "methods of operation" and "systems" as well as "ideas" uncopyrightable. The First Circuit applied the dichotomy in the statute and case law, finding that the commands are on the uncopyrightable side of the line (which

---

representing over 1000 independent software developers, consultants and software providers; Amicus Brief of 81 Distinguished Computer Scientists; Brief of Amicus Curiae (PC User Groups), representing over 16,500 individual and corporate PC users, including Fortune 500 companies; Brief of Amicus Curiae of American Committee for Interoperable Systems (including members such as Sun Microsystems, Tandem Computers and AT&T Global Information Solutions); Two Briefs Amicus Curiae on behalf of 25 Distinguished Copyright Law Professors; and Brief of Amicus Curiae on behalf of the Register of Copyrights.



Lotus calls the "distinction") between "ideas," "methods of operation," and "systems," on the one hand, which are uncopyrightable, and "expression" and "description," on the other hand, which are copyrightable.<sup>7</sup>

Contrary to Lotus' position, *no circuit* has held that the menu commands that operate the program are copyrightable on *any theory*, "idea/expression" dichotomy, "abstractions" or otherwise. In fact, it was the district court in this case that departed from existing authority when it declined to follow the Ninth Circuit's holding that the menu command hierarchy of a spreadsheet is uncopyrightable under 17 U.S.C. § 102(b). In declining to follow the Ninth Circuit's decision, *Ashton-Tate Corp. v. Ross*, 916 F.2d 516, 521-22 (9th Cir. 1990), the district court stated:

In the interest of completeness and candor, I note as well that courts in one circuit are not bound by the decisions of other circuits.

*Borland II*, 799 F. Supp. at 220, Pet. App. at 136a. Apparently lacking the "completeness and candor" of the district court, the Lotus petition does not even mention the *Ashton-Tate* decision.

The only other circuit (the Tenth Circuit) to have directly considered the copyrightability of menu commands vacated a district court decision holding that menu commands are copyrightable and provided instructions for further consideration of the issue. *Gates Rubber Co. v. Bando Chem. Indus., Ltd.*, 9 F.3d 823, 843-44 (10th Cir. 1993). On remand, the district court held that the menu commands were *uncopyrightable*, citing the First Circuit's *Lotus* decision approvingly. *Gates Rubber Co. v. Bando Chem. Indus., Ltd.*, No. 92-S-136 (D. Colo. filed June 12, 1995).

<sup>7</sup> It is true, of course, that a computer program might be described as a "method of operating" the computer and, as a consequence, Congress was required to expressly include "computer programs" in the copyright statute to ensure that computer programs would be copyrightable. But Congress did not amend the copyright statute to make menu commands, buttons, switches, and similar methods of operating a program copyrightable. On the contrary, § 102(b) of the Copyright Act plainly states that "methods of operation" are uncopyrightable. Lotus would have the courts make law that contradicts the express language of the statute.

Lotus' assertion that "[i]t is now legal to copy menus in the First Circuit but not in the Fifth, Ninth or Tenth," Pet. at 28, is simply preposterous.

Similarly, the Lotus petition claims that a particular Fifth Circuit decision "relied heavily" on the district court's analysis. Pet. at 23, citing *Engineering Dynamics, Inc. v. Structural Software, Inc.*, 26 F.3d 1335, 1348-49 (5th Cir. 1994). Incredibly, the Lotus petition fails to point out that following publication of its initial opinion, the Fifth Circuit was deluged by requests for rehearing from software and semiconductor companies, computer scientists and user groups, and subsequently issued a *supplemental opinion*. The Fifth Circuit limited its first decision to the facts of that case and specifically disclaimed reliance on the rationale that underlies the district court's opinions at issue here. *Engineering Dynamics, Inc. v. Structural Software, Inc.*, 46 F.3d 408, 409 (5th Cir. 1995). It is unfathomable that Lotus could assert that neither the Second Circuit nor the Fifth, Ninth and Tenth Circuits "has suggested that the central tenets of [the district court's] analysis have been anything but correct." Pet. at 22.

In short, there is simply no conflict among the circuits on the protection of menu commands and similar methods used to operate computer programs. Furthermore, given the subject matter that is really at issue in this case and the narrow basis of the First Circuit's opinion, virtually any comment by this Court on the copyrightability of code or program structure in a review of this case would be *dicta*. There are, on the other hand, a variety of cases currently proceeding through the lower courts that directly address those issues (discussed in Section IC below).

## STATEMENT OF THE CASE

There is good reason why the district court's opinions in this case generated such widespread attention and controversy. In the usual software copyright case, the defendant is alleged to have copied either the text ("code") or structure of the plaintiff's computer program, or the way the plaintiff's program looks on the computer screen when it is executing. No such copying occurred here. Lotus did not even allege any copying of its code or code

structure, and even the district court found that the programs looked different on the computer screen. *Borland II*, 799 F. Supp. at 220, Pet. App. at 137a.

Users operate the program at issue here through a series of commands—common English words—arranged in hierarchies called “menus.” 49 F.3d at 809, Pet. App. at 4a. As the First Circuit opinion explains, the precise facts of this case are critical to its disposition. The menu commands of the product at issue do not function as labels or explanations for any buttons or switches used to operate the program. Rather, the menu commands *themselves* are used to operate the program in much the same way that the buttons on a video cassette recorder (VCR) operate that machine. While such utilitarian methods of operation are perhaps patentable under the patent laws, this is not a patent case since Lotus obtained no patent on this or any other relevant aspect of Lotus 1-2-3.

As the First Circuit found, “the entire Lotus menu command hierarchy is essential to operating” the program. 49 F.3d at 815-16, Pet. App. at 15a-22a. If a user types “COPY” or “C,” the program copies. Typing “PRINT” or “P” causes the computer to print. Typing more complex sequences of commands in the hierarchy executes other operations of the spreadsheet. There is other text attendant to a computer program—not copied by Borland—that communicates with the user or that provides information to the user. Such text includes books about the program, manuals, on-screen “help” text, and other textual material that *does* not operate the computer. In contrast, the words of the Lotus menu command hierarchy are literally the method of operating the spreadsheet program. These limiting facts are crucial to the disposition of this petition.

#### A. Factual Background

##### 1. The Lotus Product

The development of the Lotus menu command hierarchy is well chronicled in the Lotus petition. According to the affidavit

submitted with the petition<sup>8</sup> the commands were organized “hierarchically,” and the manner depicted in a “menu tree,” so that “the selection of one command option from the first level menu could lead in turn to another array of command options on a second level menu (or ‘submenu’) and so on.” Kapor Aff. at ¶¶44-45, Pet. App. at 287a. The Lotus developer drew a firm line between the menu choices in the hierarchy (the “command options”) and the explanation of those command options. According to the affidavit, for example, the “long prompts” (or “screen help” text) are intended to provide “information to the user” and “explanations,” while the menu commands are directed toward “performing a particular task.” Kapor Aff. at ¶¶44, 101, Pet. App. at 287a, 296a.

As set forth in great detail in the lower court record, the Lotus spreadsheet was a great success, but that success had little to do with the menu command hierarchy. Rather, when the IBM PC was introduced in August 1981, the Lotus developers, according to the district court, “exploited this opportunity” by designing the technical aspects of their spreadsheet product to take advantage of the technological advances of the IBM PC over pre-existing computers. *Paperback*, 740 F. Supp. at 65-66, Pet. App. at 231a.

The menu command hierarchy was simply not a qualitatively significant part of the product *at the time of its introduction*,

---

<sup>8</sup>It should be noted that the Kapor Affidavit was *not* prepared in support of Lotus’ position in the *Borland* case. Rather, the affidavit was prepared for Lotus in the earlier *Paperback* case in which the defendant copied Lotus’ entire screen display, not merely the command words. Subsequent to the *Paperback* decision, as the record below indicates, the author of the affidavit (who is the principal designer of the Lotus product) made it clear in Congressional testimony that he does not believe it is beneficial to the industry to extend copyright protection to individual elements of a screen display—e.g., the menus. That, in his view, would constitute “overprotection” that is “pernicious” and “stifling.” See Brief of Amicus Curiae Software Entrepreneur’s Forum, filed in the First Circuit on Dec. 23, 1993, at 6.



either from Lotus' viewpoint or the viewpoint of users.<sup>9</sup> The exact words and order of the Lotus menu command hierarchy were not important to the product's initial success, but they *became* vitally important to the success of later versions of Lotus' product and spreadsheets offered by Lotus' competitors for two reasons. First, as Judge Boudin explains (and the record below demonstrates in detail), users invested their own time, money and energy in learning the Lotus commands as keystroke combinations to operate the spreadsheet, just as users operate a typewriter to produce readable text by touch typing on the standard QWERTY keyboard. 49 F.3d at 819-21, Pet. App. at 24a-26a. Second, as both the majority opinion and Judge Boudin explain,<sup>10</sup> users automate those steps by creating "macros," computer programs written by users themselves.

Lotus' own documents and the record below demonstrate that while Lotus' product initially became a success because it was technologically superior to its early competition, it later maintained its share because, as Judge Boudin noted, the user's investment in learning the method of operation of the Lotus product and the creation of macros "locked in" those users who first selected Lotus over its inferior early competition.<sup>11</sup> Therefore, unless a new entrant with a superior product in the spreadsheet market could compete for the business of the vast majority of computer users who initially chose Lotus, competition would be limited solely to new spreadsheet users, a minor portion of the market. In short, there would be little, if any, business for which to compete.

<sup>9</sup>See, e.g., Raburn Decl. at ¶14, RE 3: *Borland III*, 831 F. Supp. at 213, Pet. App. at 86a-87a.

<sup>10</sup>49 F.3d at 809-10, 819-21, Pet. App. at 4a, 24a-27a. See also *Paperback*, 740 F. Supp. at 64-65, Pet. App. at 228a; *Borland II*, 799 F. Supp. at 213-14, Pet. App. at 110a; *Borland IV*, 831 F. Supp. at 227, Pet. App. at 31a.

<sup>11</sup>Ex. 38 (Houdini Analysis of Competitive Products) at L047694, 1st Cir. App. 1196. As a result of macros, 1-2-3 became, in the words of Lotus' own documents, "entrenched." Ex. 39 (Review of Excel) at L046265, 1st Cir. App. 1198.

## 2. The Borland Product

The Lotus petition attempts to dismiss Borland's product as a "clone" or imitator, but the record below demonstrates otherwise. The Borland product was first introduced in 1989 and won every major award for spreadsheet excellence given in the software industry. The Borland product invariably ranked higher than the Lotus product in head-to-head reviews and user comparisons, including those conducted by Lotus.<sup>12</sup>

Not only was the Borland product superior from technological and performance perspectives, but it also employed a new screen display that was different in every respect (save the command words) from the Lotus products at issue in this case. See Borland S.J. Brief (Dkt. No. 141) at 116-34. The Borland product had its own menu command hierarchy designed specifically to take advantage of its superior technological features and a different screen display. But the Borland product also provided, as an alternative, an enhanced version of the Lotus menu command hierarchy so that users who were locked into the earlier menus could operate the new product without relearning their spreadsheet skill set or rewriting their macros.<sup>13</sup>

## B. Proceedings in the District Court

Throughout the proceedings below, the district court resisted every attempt Borland made to secure prompt appellate review of the district court's controversial extension of copyright law. The district court's earlier *Paperback* decision had not been appealed because Lotus settled the case following the district court's ruling and the defendant went out of business.

The district court issued its first opinion in this case on March 20, 1992, indicating that it intended to continue to employ its widely criticized methodology from the *Paperback* case. *Borland I*, 788 F. Supp. at 89-90, Pet. App. at 163a. On July 31,

<sup>12</sup>See Ex. 1 and 2, 1st Cir. App. 1134; Borland S.J. Brief (Dkt. No. 141) at 2 n.3.

<sup>13</sup>P. Kahn Tr. at 65-66, RE 10-11; Bosworth Tr. at 148, 1st Cir. App. 748; Ex. 27 at L034481, 1st Cir. App. 1178.

1992, the district court published its *Borland II* decision and an accompanying procedural order holding that "[t]he menu commands and menu hierarchy of Lotus 1-2-3 have expressive aspects and are copyrightable." Order Regulating Jury Trial, finalized Sept. 30, 1992 (Dkt. No. 232), at 18. However, the district court said that it could not determine the "precise scope of Borland's infringement" without further trial proceedings, *Borland II*, 799 F. Supp. at 221, Pet. App. at 138a, and set a trial to begin on November 2, 1992.

Immediately following the district court's *Borland II* decision, Borland pulled the Lotus menus from its products. On three separate occasions, Borland moved to certify for interlocutory appeal the question of the menu command hierarchy's copyrightability. However, the district court denied all three of Borland's motions, greatly extending the proceedings.<sup>14</sup> Ultimately, Borland argued in favor of enjoining its own product so that appellate review would be available pursuant to the injunction statute, 28 U.S.C. § 1292(a). On August 19, 1993, the district court entered a permanent injunction against the continued sale of Borland's product and an appeal was promptly taken to the First Circuit.

The district court refused to stay either damages discovery or a damages trial pending the First Circuit's decision on liability. As a result, the damages phase of the case, which was eventually terminated by the First Circuit's reversal, cost Borland millions of dollars in legal and expert fees. Although the district court was eventually reversed, the series of district court opinions, followed by the entry of a permanent injunction, had a devastating effect on Borland. For example, as the record below reflects, Borland's stock, which traded at 86¼ in January, 1992, prior to the first of the district court's opinions in the case, fell to 12¼ in September 1993, following entry of the injunction. Borland was required of financial necessity to sell its spreadsheet products to another company and is no longer a competitor of Lotus, which is now owned by International Business Machines Corp. (IBM). Any

<sup>14</sup>See Tr. of Sept. 23, 1992 Hearing at 54, 1st Cir. App. 18; Tr. of Oct. 16, 1992 Hearing at 15-17, 1st Cir. App. 24-25; Tr. of Aug. 19, 1993 Hearing (Dkt. No. 406) at 42-43.

further appellate proceedings, even if successful from Borland's legal viewpoint, can be exploited by IBM to further damage Borland competitively.

## REASONS FOR DENYING THE PETITION

### I. THE CIRCUITS ARE IN AGREEMENT THAT WORDS USED AS BUTTONS TO OPERATE A PROGRAM, UNLIKE THE PROGRAM ITSELF, ARE UNCOPYRIGHTABLE.

#### A. The District Court Erroneously Applied the "Abstractions" Test to Uncopyrightable Subject Matter.

The Lotus petition attempts to inject this case into the ongoing debate in the case law involving the extent to which the "structure" of a computer program should be protected by copyright. But extending copyright protection to menus and similar methods of operation has far greater legal and economic consequences than extending copyright to the structure of a computer program. The legal argument that at least some aspects of a program's structure might be copyrightable is a familiar one. A computer program, as defined in 17 U.S.C. § 101 (a "set of instructions" used "to bring about a certain result"), is a "literary work." The literal elements of the program, the source code and object code, are copyrightable in the same way that the literal text of a play would be.

Within the genre of *artistic* literary works like plays, courts have created the "paraphrasing" doctrine, *see* Pet. at 10, also known as the concept of "comprehensive non-literal similarity," as a basis for copyright infringement. *See* David Nimmer & Melville B. Nimmer, *Nimmer on Copyright*, § 13.03[A][1] at 13-29 (1994). If, for example, someone made an unauthorized translation into French of a play originally written in English, the authors would have a claim for copyright infringement, notwithstanding the fact that the original work (in English) and the unauthorized "copy" (in French) do not share any common literal expression. They are not substantially similar in terms of literal expression, but they are similar with respect to the "struc-



ture" of the play, each of its acts, each of its scenes, and, for that matter, the breakdown of the dialogue into sentences.

Because Congress decided to apply copyright protection to the code of a computer program (the "set of instructions"), one might argue (as Lotus does) that copyright doctrines for artistic literary works should apply to some extent to the code of a computer program, which is also a literary work. See, e.g., *Whelan Assoc., Inc. v. Jaslow Dental Laboratory, Inc.*, 797 F.2d 1222, 1233-34 (3d Cir. 1986), cert. denied, 479 U.S. 1031 (1987). Thus, if a programmer writes a program in "BASIC" and someone "translates" that program into the programming language "C," the second program is an unauthorized copy notwithstanding the absence of literal similarity at the code level. The two programs are "substantially similar" with respect to their detailed "structure." Hence, a few courts initially applied the doctrine of "comprehensive non-literal similarity" to protect the "structure, sequence and organization" of a computer program, at least to some limited extent. See, e.g., *Altai*, 982 F.2d at 702-04 (citing cases).

But application of the concept of "comprehensive non-literal similarity" to computer programs has been controversial because, unlike artistic works (such as plays), computer programs are utilitarian objects—they perform a function.<sup>15</sup> Copyright, unlike patent, is a very broad, long-lasting, easily obtainable type of protection, and Congress has always taken care to ensure that easily obtainable copyright protection would not be available for the utilitarian or functional aspects of works.<sup>16</sup> As Professor Miller, formerly a member of CONTU, stated:

The end purpose of a computer program is to achieve a utilitarian result, i.e., the computer's performance of logical operations in a way that produces the desired practical consequence. One cannot compare, therefore, the underlying

<sup>15</sup> *Altai*, 982 F.2d at 704; Paul Goldstein, *Copyright*, § 2.15 at 195 (1989) ("Goldstein Treatise").

<sup>16</sup> See, e.g., H.R. Rep. No. 1476, 94th Cong., 2d Sess. 54-55 (1976), reprinted in 1976 U.S.C.A.N. 5659, 5667-68.

processes of a computer program with, say, the underlying plot structure of a novel or a screenplay of a movie. This, of course, is the distinction recognized by the Supreme Court long ago in the seminal decision of *Baker v. Selden*, 101 U.S. 99 (1879).

Kenneth A. Liebman, et al., *Back To Basics: A Critique Of The Emerging Judicial Analysis Of The Outer Limits Of Computer Program "Expression,"* 2 Computer Law., December 1985 at 1, 8 (quoting Arthur Miller). Professor Miller, now Lotus' counsel, summarized this concern best when he opined in a declaration in another case: "The creativity, ideas and utilitarian aspects of a copyrighted work must look elsewhere for legal protection."<sup>17</sup>

The earliest attempt to apply "non-literal similarity" to computer programs, the Third Circuit's *Whelan* decision, resulted in protection for virtually all of the program's structure. *Whelan*, 797 F.2d at 1238. The Second Circuit in *Altai* rejected the *Whelan* test because the *Whelan* test made too much copyrightable. *Altai*, 982 F.2d at 705-06. Under the *Whelan* test, the "function" of the computer program is the work's protectable idea, and "everything that is not necessary to that purpose or function would be part of the [protectable] expression of the idea." *Whelan*, 797 F.2d at 1236. Under the "abstractions" test formulated by the *Altai* court, far fewer aspects of code structure are protected by copyright. The leading post-*Altai* decision, the Tenth Circuit's *Gates Rubber* case, limited the protection of code structure even further by requiring the lower courts specifically to consider the proscriptions of § 102(b) in applying the abstractions and filtrations formulations. *Gates Rubber*, 9 F.3d at 833, 836.

The test formulated by the district court in this case was based on *Whelan* rather than *Altai*, and, for that reason, the *Altai* court

<sup>17</sup> In declarations in an earlier case in which Professor Miller was the retained expert for the party accused of infringement, Professor Miller opined on the CONTU deliberations and the role of copyright in protecting computer programs. Those declarations can be found in the record below as exhibits to the Reply Brief of Defendant/Appellant Borland International, Inc., filed in the First Circuit on March 1, 1994. The quoted portion in the text is from the second declaration at ¶29.

rejected the lower court's approach as having a "corrosive effect on certain fundamental tenets of copyright doctrine." *Altai*, 982 F.2d at 712. More germane for the purposes of Lotus' petition is the fact that the district court sought to apply an abstractions-type analysis to a menu command hierarchy which, unlike a "computer program," is not copyrightable in the first place. *Borland II*, 799 F. Supp. at 216-19, Pet. App. at 128a-135a.

The "abstractions" test was initially applied to the text of plays—clearly copyrightable subject matter. Similarly, the leading cases cited by Lotus—*Whelan*, *Altai* and *Gates Rubber*—apply the abstractions test to the code of "computer programs" which Congress expressly has said is copyrightable. By contrast, the district court here applied its own variant of the abstractions test to the method of operating the program, the menu commands. The Lotus petition deftly slides over this critical distinction. Compare Pet. at 14-16 (describing the application of the test by various courts of appeals to the "program") with Pet. at 17 (discussing the district court's application of the test to so-called non-literal elements).

As the First Circuit opinion points out, the abstractions test assumes that the work at issue is copyrightable. 49 F.3d at 815, Pet. App. at 14a. Application of the "abstractions" test to the menu command hierarchy inevitably led the district court to find *something* in the menu command hierarchy copyrightable, *id.*, and the district court concluded that the specific commands and order chosen by Lotus were protected by copyright. *Borland II*, 799 F. Supp. at 217, Pet. App. at 131a. In other words, as the First Circuit explained, the district court's test devolved to a question of whether choices exist for the subject matter at issue. 49 F.3d at 811, 816, Pet. App. at 6a, 17a. Under the district court's methodology, if there are choices, the subject matter at issue is copyrightable. But the mere existence of a choice does not turn uncopyrightable subject matter into copyrightable "expression." One might equally argue that the engines of a Ferrari and a Volkswagen embody different "expressions" of the process of internal combustion, or that the QWERTY and Dvorak keyboards are different ways of "expressing" the means by which the user operates a typewriter.

Lotus responds to this reasoning by arguing that "computer programs," unlike car engines and keyboards, are copyrightable. Pet. at 29. Had the district court applied its abstractions test to the *computer program*, Lotus' argument would be germane. But the district court applied its test to the menu command hierarchy, not to the program. *Borland II*, 799 F. Supp. at 216-19, Pet. App. at 128a-135a.

Had the district court applied an abstractions test to the *programs* at issue, it would have quickly found no code or "structural" similarity on any level whatsoever—nonliteral or otherwise. Indeed, there was no allegation in this case that Borland copied or even had access to the "structure" of the Lotus program. Lotus did not produce its code in discovery, neither party introduced the code of its program into evidence at trial, and there is every reason to believe that the Lotus program and the Borland program have vastly different structures to perform the same functions. In short, even after application of the "abstractions" test to the program, the district court would still be faced with the issue of whether the menu commands are copyrightable. That issue can only be resolved by reference to the statute and case law dealing with utilitarian works. In any event, the issues of nonliteral similarity and copyrightability of code structure could not properly be reviewed by this Court in this case, since the narrow issue present here does not raise those broader issues.

#### **B. The District Court's Decision to Extend Copyright to the Words That Operate the Program Ran Afoul of Both the Statute and the Case Law.**

The First Circuit's reversal of the district court rests on the familiar language of § 102(b):

In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work.

Although the Lotus petition at one point claims that the First Circuit's opinion rejected the idea/expression dichotomy, Pet. at



23, the earlier discussion in the Lotus petition correctly points out that § 102(b) is, in fact, the legislative *embodiment* of the idea/expression dichotomy. Pet. at 10. In short, the First Circuit embraced the demarcation in § 102(b) between copyrightable subject matter and uncopyrightable "methods of operation," "systems," and "ideas" which must look to patent law for protection. It was the district court that altered that fundamental demarcation. The First Circuit merely followed the "line" between copyrightable and uncopyrightable subject matter previously established by Congress and this Court in *Baker v. Selden*.

Confronted with the plain language of § 102(b) proscribing copyright protection for methods of operation, the district court limited the statute to abstractions. In the view of the district court, whenever any words are attached to § 102(b) subject matter (such as "processes" or "methods of operation"), the words become copyrightable. See *Borland I*, 788 F. Supp. at 91, Pet. App. at 167a; 49 F.3d at 816, Pet. App. at 17a. As the First Circuit pointed out, limiting uncopyrightable § 102(b) subject matter to abstractions moves the line established long ago in *Baker v. Selden* between copyrightable and uncopyrightable subject matter. For more than 100 years, until the district court's opinion, copyright law was grounded on the proposition that the barest words that state a system or operate a machine (e.g., "move," "copy"), as opposed to a description of those operations (e.g., the long prompts) are uncopyrightable. The district court's opinions moved the "line" between copyrightable and uncopyrightable subject matter established in *Baker v. Selden*, and that is what produced the enormous public outcry.

In *Baker*, the plaintiff, Charles Selden, obtained a copyright on a pamphlet that explained a systematic approach to bookkeeping. The pamphlet contained a complex series of ledgers or forms, like the various screen displays in the Lotus user interface. The Selden forms each contained grids, columns, and various alternative short textual descriptive "headings" or "captions" (such as "Balance Forward") like the Lotus menu commands.

The defendant, Baker, published forms similar in headings and arrangement to those of Selden. Selden sued Baker for copyright infringement because of the similarity, arguing—as Lotus argues

here—that there was "original expression" in the selection, ordering and arrangement of the headings and columns of the ledgers each contained in his copyrighted pamphlet. See *Baker v. Selden*, 101 U.S. at 101.

Manifestly, since the words on Baker's forms were different from those on Selden's, this Court could have found for Baker solely on the ground that the textual labels were not substantially similar. But this Court did not take that route. Stating the principal issue in the case as whether Baker could use "similar ruled lines and headings, or ruled lines and headings made and arranged on substantially the same system, without violating [Selden's] copyright," *id.*, at 101, this Court held that Selden's ledgers, including their column arrangement and textual headings, were not copyrightable at all—and could be copied verbatim. *Id.* at 107.

As the *Altai* court observed, 982 F.2d at 704, the holding of *Baker* that methods of operations and systems are not copyrightable is *not* restricted to *pure abstractions*:

[T]he holding in *Baker* goes farther. The [Supreme] Court concluded that those aspects of a work, which "must necessarily be used as incident to" the idea, system or process that the work describes, are also not copyrightable. 101 U.S. at 104.

The First Circuit relied upon *Baker* in the same manner as did *Altai*. It relied upon Baker's limitations on the scope of copyright to conclude that the commands used to operate the Lotus program were not copyrightable. 49 F.3d at 816–17, Pet. App. at 18a. Indeed, the district court's limitation on § 102(b) not only ran afoul of *Baker v. Selden* but was also at variance with the unique facts of this case. Here, as the First Circuit explained, the words at issue are more fundamental to the operation of the program than even labels on buttons would be. 49 F.3d at 817, Pet. App. at 18a–19a. Here, the words of the menu command hierarchy are "essential to operating" the program and, hence, are part of the method of operation. *Id.* at 18a. As the First Circuit explained, "it would be impossible to operate [the Lotus program] without employing its menu command hierarchy." *Id.* at

19a. The holding of the First Circuit, narrowly tailored to the facts before it, is wholly consistent with similar cases in other circuits.

**C. There Is No Conflict Among the Circuits That Menu Commands and Similar Methods of Operation Are Uncopyrightable.**

Lotus argues that the First Circuit's reasoning is contrary to the law in other circuits, and that review of this case is needed to resolve a conflict in the circuits. To the contrary, there is no such conflict on the narrow issue actually presented here.

1. *The Ninth Circuit.* Lotus cites two cases for an alleged conflict between the First and Ninth Circuits. First, *Johnson Controls, Inc. v. Phoenix Control Sys., Inc.*, 886 F.2d 1173, 1175-76 (9th Cir. 1989), is a code structure case and not pertinent here at all. Second, Lotus relies upon *Brown Bag Software v. Symantec Corp.*, 960 F.2d 1465, 1477 (9th Cir.), *cert. denied sub. nom. BB Asset Mgmt. Inc. v. Symantec Corp.*, 113 S. Ct. 198 (1992), as allegedly creating a conflict. This is misplaced. That Ninth Circuit opinion specifically affirmed the lower court's holding that the menus at issue were "unprotectable under copyright." *Id.* at 1472. (While there is ambiguous *dicta* elsewhere in *Brown Bag* which Lotus cites for the proposition that "menus and keystrokes" are copyrightable, which the First Circuit also noted, *see* 49 F.3d at 819 n.14, Pet. App. at 22a n.14, that *dicta* does not alter the Ninth Circuit's ultimate opinion.)

In fact, a prior Ninth Circuit decision—not cited by Lotus—explicitly held that the menu commands of a spreadsheet software product were uncopyrightable. *Ashton-Tate Corp. v. Ross*, 916 F.2d 516 (9th Cir. 1990). In *Ross*, the plaintiff alleged that he had designed and given Ashton-Tate a complete menu hierarchy, including numerous submenus, which Ashton-Tate incorporated into its "Full Impact" spreadsheet product without compensating him. Unlike this case, Ross' list was handwritten, and had been developed before any computer code was written. However, like the Lotus 1-2-3 menu tree, Ross' tree contained both main menus and submenus. (A copy of Ross' actual menu

hierarchy was included in the district court record in this case. *See Borland II*, 799 F. Supp. at 220, Pet. App. at 136a.)

The *Ashton-Tate* district court ruled against Ross, finding that he was not entitled to compensation because the spreadsheet menu hierarchy was not entitled to copyright protection. Citing 17 U.S.C. § 102(b), the District Court held that Ross' "list of labels for user commands . . . is not protected under federal law." *Ashton-Tate Corp. v. Ross*, 728 F. Supp. 597, 602 (N.D. Cal. 1989). On appeal, Ross renewed his argument, the very same argument to justify copyrightability (*i.e.* the presence of choices) advanced by the district court here. Ross argued that his menu command hierarchy evidenced

numerous decisions by the authors about the ordering of the commands and their arrangement in the user interface. The fact that the authors of these design documents chose the order and groupings displayed, out of a nearly infinite number of possibilities, constitutes creative authorship.

Appellants' [Ross'] Opening Brief on Appeal at 25, 1st Cir. App. 1168. The Ninth Circuit confronted this argument directly and rejected it, stating that "[t]his argument is meritless for the reasons given in the district court's order, 728 F. Supp. at 602. The list simply does not qualify for copyright protection." 916 F.2d at 521-22.

Lotus does not mention *Ashton-Tate*, perhaps because the district court explicitly declined to follow it. *Borland II*, 799 F. Supp. at 220, Pet. App. at 136a. By reversing the district court and agreeing with *Ashton-Tate's* conclusion, the First Circuit removed any conflict with the Ninth Circuit, rather than creating one.

2. *The Tenth Circuit.* Lotus argues, and the First Circuit suggested, that the court's holding was in conflict with *dicta* in a footnote in *Autoskill, Inc. v. National Educ. Support Sys., Inc.*, 994 F.2d 1476, 1495 n.23 (10th Cir.), *cert. denied*, 114 U.S. 307 (1993). A careful examination of *Autoskill* and subsequent Tenth Circuit authority reveals that there is no such conflict worthy of review at this time. Footnote 23 in *Autoskill* appears to hold that, for the purposes of a preliminary injunction, the district court did



not improperly enjoin a computer program where a student/user selected responses to the program's queries "by pressing the 1, 2, or 3 keys." 994 F.2d at 1495 n.23. The First Circuit noted this *dicta* and declined to "follow" the reasoning of this footnote. 49 F.3d at 813, 818-19, Pet. App. at 12a, 21a-22a.

To the extent that the *Autoskill* footnote bears on the issues of copyrightability of menus and was rejected by the First Circuit, it had *already been rejected* by the Tenth Circuit itself. In *Gates Rubber*, 9 F.3d 823, the Tenth Circuit limited its prior holding in *Autoskill* because that case only involved the review of a preliminary injunction order. *Id.* at 841. The Tenth Circuit vacated the *Gates Rubber* district court's finding that computer menus were copyrightable, and remanded the case to that district court for further consideration and analysis under the appropriate legal standards. *Id.* at 843-44.

The uncopyrightability of computer menus in the Tenth Circuit was confirmed by the very recent decision by the *Gates Rubber* district court, after the Tenth Circuit's remand. *Gates Rubber Co. v. Bando Chem. Indus., Ltd.*, No. 92-S-136 (D. Colo. filed June 12, 1995). Using the analysis ordered by the Tenth Circuit, the District of Colorado found that the menus of that program were uncopyrightable. Slip op. at 6-7. It saw no conflict among the circuits. To the contrary, the Colorado court approvingly cited the First Circuit's opinion in *Lotus v. Borland* without noting any conflict. *Id.* at 7. To the extent that *Gates Rubber* still poses any issues worthy of review, this Court can ultimately grant review of that case.<sup>18</sup>

<sup>18</sup> A few district courts have protected the menu aspects of screen displays by generally focusing on artistic and stylistic aspects of display, rather than the words themselves. For example, in *Digital Comm. Assoc., Inc. v. Softklone Distrib. Corp.*, 659 F. Supp. 449, 460 (N.D. Ga. 1987), the district court protected the "highlighting" and "capitalizing" of certain menus only because they "have no relationship to the functioning . . . of the computer program." Similarly, *Manufacturers Technologies, Inc. v. CAMS, Inc.*, 706 F. Supp. 984, 995-998 (D. Conn. 1989), provided protection to only the three screens that were not limited by functionality and denied copyright protection to the words and format of most menus. Where only the methods of operation were

3. *The Fifth Circuit.* Lotus also relies upon *Engineering Dynamics*, 26 F.3d 1335 (5th Cir. 1994), *supplemented on pet. for reh'g*, 46 F.3d 408 (5th Cir. 1995). Initially, *Engineering Dynamics* was the only court of appeals case which, like the district court opinions here, purported to apply an "abstractions"-like test *directly* to a "non-literal element." That particular non-literal element was the input formats to the computer program, and not a method of operating the program. Therefore, § 102(b) did not play any role in the court's decision. Indeed, the court did not base its reasoning on any analysis of § 102(b) or of methods of operation.

As discussed above, following the publication of the original opinion in *Engineering Dynamics*, the Fifth Circuit was deluged with requests for rehearing. It issued a supplemental opinion (not cited by Lotus) which greatly if not completely undercuts Lotus' arguments. The Fifth Circuit's supplemental opinion rejected the assertion that it protected the user formats in that case because there were "numerous ways the input formats could be organized." It instead stated that "[t]he panel did not say that in any case involving user interface the fact that the 'author' has selected from among possible formats is dispositive." 46 F.3d at 409. This is consistent with the First Circuit's views on one of the issues in dispute here, namely whether the availability of "expressive choices" in designing the menu command hierarchy makes the menus copyrightable. See 49 F.3d at 816, Pet. App. at 17a. Significantly, since the Fifth Circuit remanded that case for further proceedings, this Court can eventually review *Engineering Dynamics* if those proceedings result in any real conflict with the First Circuit.

4. *The Second and Third Circuits.* Finally, Lotus argues that this case conflicts with the Second and Third Circuit opinions in *Altai* and *Whelan*. As explained above, those cases involved the non-literal copying of code structure, rather than the "method of operation" issues involved here, and hence do not conflict with

---

involved, district courts have declined to protect computer menus under the copyright laws. See *Mitek Holdings, Inc. v. Arce Eng'g Co.*, 864 F. Supp. 1568, 1579-80 (S.D. Fla. 1994).

the narrow issues presented by this case. Moreover, Lotus is incorrect that the First Circuit "rejected" *Altai's* abstraction-filtration-comparison test. The First Circuit did not reject *Altai*; to the contrary, it held that "the *Altai* test may provide a useful framework for assessing the alleged nonliteral copying of computer code." 49 F.3d at 815, Pet. App. at 14a. That issue is not present here, and could not properly be dealt with by this Court upon review. Finally, while *Altai* and numerous other courts have harshly criticized *Whelan*, see 982 F.2d at 705-06, that hardly makes *this* a suitable case to review the viability of *Whelan*. Indeed, since *Altai* is still pending before the Second Circuit, this Court can grant review of that case to address the *Altai/Whelan* debate.

## II. THE FIRST CIRCUIT OPINION REMOVES THE UNCERTAINTY PRODUCED BY THE DISTRICT COURT AND UPHOLDS THE OVERALL INTELLECTUAL PROPERTY FRAMEWORK ESTABLISHED BY CONGRESS.

### A. The First Circuit Opinion Restores Clarity and Predictability to the Law.

In the First Circuit, many of Borland's amici urged reversal of the district court because of the uncertainty created by the district court's methodology and result. It is ironic in the extreme that Lotus would now petition this Court, claiming it is the First Circuit's decision that has produced uncertainty. Software developers obviously need clear rules to enhance productivity. Prior to the district court's decisions, developers and their counsel, relying on § 102(b), *Baker v. Selden* and *Ashton-Tate v. Ross*, believed that menu commands and similar methods of operating a program were uncopyrightable. The First Circuit has now restored that clarity.

The alternative offered by Lotus, a case-by-case determination of whether § 102(b) means what it says, would have a chilling effect on software development. One need look no further than the facts of this case to understand the grave difficulties such a regime would portend. Here, two years, two opinions, hundreds of pages, and millions of dollars into this case, the district court

could still not determine the "precise scope of Borland's infringement," *Borland II*, 799 F. Supp. at 221, Pet. App. at 138a, without another year and one-half of proceedings. Under the regime proposed by Lotus, new and better products that compete for the business of a competitor's customers can be brought to market legally only at the cost of one's company. In rejecting such a regime, the First Circuit has restored clarity to at least a portion of the overall intellectual property protection framework established by Congress. Equally important is the fact that the First Circuit's opinion restores the long-standing demarcation between copyright and patent law.

Copyright is broad, long-lasting, easily obtainable protection. Copyright protection is obtained for copyrightable elements of a computer program merely by making a deposit of any copyrightable subject matter (such as code). See Copyright Office Circular 61, *Copyright Registration for Computer Programs*, at 2, 1st Cir. App. 1170. There is no examination procedure. The copyright lasts for approximately 75 years. Because copyright protection is so easy to obtain, and lasts so long, it was neither intended for, nor is it suited for, the granting of government-sanctioned monopolies for methods of operation. See, e.g., *Goldstein Treatise*, § 2.3.1 at 78, § 2.15.2 at 207; Paul Goldstein, *Infringement of Copyright in Computer Programs*, 47 U. Pitt. L. Rev. 1119, 1123-24 (1986). The monopoly on a menu command hierarchy or similar "method of operation" has far greater ramifications than even the monopoly on the "structure" of a program. Protecting code structure has no preclusive effect on the program's users. However, precluding a competitor's product from offering another's method of operation means that users will lose their investment in the skill set necessary to implement that method of operation if they switch to a competitor's product. 49 F.3d at 821, Pet. App. at 26a-27a.

Such a broad government-sanctioned monopoly must be secured, if at all, through the patent system. Patents on "methods of operation" are difficult to obtain and last a relatively short period of time (20 years or less). Patent applications must state the invention, describe the prior art, and set forth the claims for protection clearly and specifically. There is a complex examination process to ensure that the patentee will be contributing



something new to the state-of-the-art (i.e., something novel and non-obvious, an advancement over the prior art, etc.), as the quid pro quo for the grant of monopoly. None of these safeguards are present in the copyright system established by Congress because it was (and is) not contemplated that the scope of copyright protection is tantamount to that of patent. Unless § 102(b) is recognized for what Congress intended it to be, the copyright law would afford over-extensive protection to works by applying only the most minimal level of scrutiny. The First Circuit's opinion is consistent with the intellectual property framework established by Congress and supports its rationale.

In short, Lotus lost this case because it tried to secure patent-type protection without satisfying the patent requirements of novelty, examination and contribution to the prior art. If, like the plaintiff in *Baker v. Selden*, Lotus cannot meet these requirements, or if it chooses not even to try, it should not be able to claim the same scope of protection through copyright law. That, as the *Baker* Court observed, "would be a surprise and a fraud upon the public. That is the province of letters-patent not of copyright." 101 U.S. at 102.

**B. Both Copyright and Patent Remain Sources of Strong Protection, Providing Enormous Incentives for Authors and Inventors of Software Products to Innovate.**

Lotus' suggestion that the First Circuit's ruling "could serve to roll back the scope of protection for computer programs generally," Pet. at 29, is wholly unfounded. Copyright protection remains a powerful and sufficient incentive for the development of new software. Copyright protects against piratical copying of object code. Copyright protects against appropriation of source code, either literally or by paraphrasing. Perhaps, in appropriate circumstances, copyright also protects against copying the detailed "structure" of another's program.

Nor does the First Circuit's opinion call into question the screen display portion of a "user interface." Compare Pet. at 28. As the *Altai* court observed, copyright protection for screen displays does not depend on the protection of "non-literal elements" of the computer program. Rather, copyright protects

against the unauthorized reproduction of "certain types of screen displays," that are "copyrighted separately as an audiovisual work." *Altai*, 982 F.2d at 703. The First Circuit did not even remotely suggest that screen displays are uncopyrightable. Rather, its holding that menu commands are uncopyrightable does not interfere with the protection of screen displays "because the way the screens look has little bearing on how users control the program." 49 F.3d at 816 & n.10, Pet. App. at 16a & n.10.

Menu commands and similar methods of operating the program continue to be protected by the patent law, as they always have been. The record below contains several examples of menu command hierarchies, including those of IBM (Lotus' new owner) that are protected by utility patents. Borland S.J. Brief, Exs. 16 & 23 (U.S. Patents Nos. 4,989,141, 4,611,306). Indeed, the U.S. Patent and Trademark Office has recently announced it is changing the rules for patentability of software, making it even easier to obtain software patents.<sup>19</sup>

## CONCLUSION

This Court has made it clear that only Congress may redraw the balance between private monopoly and public access. The courts are required to defer to Congress "when major technological innovations alter the market for copyrighted materials." *Sony Corp. v. Universal City Studios, Inc.*, 464 U.S. 417, 431 (1984). If Congress has not expressly chosen to expand the scope of copyright protection, it is not the job of the courts to do so. On the contrary, "[i]n a case like this, in which Congress has not plainly marked our course, we must be circumspect in construing the scope of rights created by a legislative enactment which never contemplated such a calculus of interests." *Id.* The district court

<sup>19</sup>See U.S. Department of Commerce, Press Release No. 95-21, *Software Patent Guidelines Released Today*, 6/1/95, 60 Fed. Reg. 28,778 (proposed June 2, 1995); B. Rosewicz, *Patent Office Acts to Clarify Software Rules*, Wall St. J., June 2, 1995, at A1; U.S. Department of Commerce, Press Release No. 95-18, *USPTO to Develop Guidelines to Protect Software Inventions*, 3/30/95; M. Betts, *Feds to ease software patent guidelines*, Computerworld, April 17, 1995, at 20.

in this case plainly thought that it was empowered to create new law in "uncharted" territory and to "draw the line between copyrightable and non-copyrightable elements of computer programs." *Borland I*, 788 F.Supp. at 90, Pet.App. at 165a; *Paperback*, 740 F.Supp. at 53, Pet.App. at 206a. In so doing, the district court usurped the role of Congress. The First Circuit corrected this error.

Twenty years ago, the first personal computers had no screens or keyboards; the users operated the machines by pressing buttons or switches on the front of the machines. No one would ever claim that such buttons were copyrightable. Twenty years from now, users will operate personal computers with spoken words, and without any physical buttons or keyboards. It is inconceivable that anyone could claim that such spoken methods of operation will be copyrightable. At the intermediate stage of technology relevant here, Lotus used typed words as the buttons or switches to operate its spreadsheet program. Those words are no more copyrightable than physical buttons were twenty years ago, or than spoken commands will be twenty years from now.

For the foregoing reasons, the petition for a writ of certiorari should be denied.

Respectfully submitted,

GARY L. REBACK  
(Counsel of Record)  
MICHAEL BARCLAY  
ISABELLA E. FU  
WILSON, SONSINI,  
GOODRICH & ROSATI  
Professional Corporation  
650 Page Mill Road  
Palo Alto, California 94304  
Telephone: (415) 493-9300

PETER E. GELHAAR  
KATHERINE L. PARKS  
DONNELLY, CONROY &  
GELHAAR  
One Post Office Square  
33rd Floor  
Boston, Massachusetts 02109  
Telephone: (617) 542-7400

*Attorneys for Respondent*

July 6, 1995